

Rechnernutzung in der Physik

Institut für Experimentelle Teilchenphysik
Institut für Theoretische Teilchenphysik
Interfakultatives Institut für Anwendungen der Informatik

Prof. Dr. U. Husemann, Dr. V. Shtabovenko
Dr. A. Mildemberger, Dr. T. Chwalek

WS2019/20 – Blatt 1

<http://comp.physik.kit.edu>

Bearbeitungszeitraum: bis Di, 5.11.2019

Das Passwort für den Online-Status von Testaten und Auswertungen auf der Kurswebseite ist bayes (klein geschrieben). Das System wird in den nächsten Tagen freigeschaltet.

Aufgabe 1: Kennenlernen von Mathematica

MATHEMATICA kann zum einen mit einer graphischen Oberfläche zum anderen in der Kommandozeile gestartet werden. Im ersten Fall verwendet man den *Frontend*, im zweiten Fall steuert man direkt den *Kernel*. Die Verwendung von Frontend macht vor allem bei interaktiven Dokumenten (mit eventueller grafischer Ausgabe), den *Notebooks* (mit Dateierdung .nb), Sinn. Diese erlauben es, Befehle schrittweise auszuführen und ihre Wirkung unmittelbar zu verfolgen. Im Poolraum lautet der Aufruf von der Kommandozeile

```
$ mathematica NOTEBOOK.nb &
```

Ebenfalls interaktiv, aber ohne Frontend, startet man den Kernel mit:

```
$ math
```

Eine Session lässt sich mit `Exit[]`, `Quit[]` oder `Ctrl-D` beenden.

Bereitgestellte Notebooks können mit dem WOLFRAM CDF PLAYER (<https://www.wolfram.com/player>) angesehen werden. Editieren und Ausführen sind damit jedoch nicht möglich.

Eine im Editor vorbereitete Datei (im Allgemeinen mit Endung .m) kann in den Kernel eingelesen und dann verarbeitet werden. Es gibt verschiedene Möglichkeiten, MATHEMATICA in diesem Modus aufzurufen:

```
$ math < PAKET.m  
$ math -noprompt < PAKET.m  
$ math -script PAKET.m
```

Für die Bearbeitung der Aufgaben werden sowohl Notebook-Dateien als auch Textdateien mit Endung .m akzeptiert.

Auf der Homepage zur Vorlesung finden Sie die Datei `a1-intro.nb`, die Sie in Ihr Verzeichnis kopieren sollten. Öffnen Sie danach das Notebook in MATHEMATICA und arbeiten Sie es Schritt für Schritt durch. Es ist empfehlenswert, diese Aufgaben zu bearbeiten, da in dem Notebook viele Sprachelemente enthalten sind, die für die Bearbeitung der MATHEMATICA-Aufgaben wichtig sind.

Hinweise:

- Bearbeitung und Abgaben in MAPLE werden ebenfalls akzeptiert, beachten Sie aber, dass für MAPLE keinerlei Hilfestellungen durch die Tutoren erwartet werden kann.

Aufgabe 2: Sieb des Eratosthenes (*)

Das Sieb des Eratosthenes ist ein Algorithmus zur Bestimmung von Primzahlen kleiner oder gleich einer vorgegebenen oberen Grenze N . Der Algorithmus lautet dabei folgendermaßen:

- (i) Erzeuge eine Liste der Zahlen $2, 3, \dots, N$.
- (ii) Die kleinste in der Liste verbleibende Zahl ist eine Primzahl p . Streiche nacheinander alle echten Vielfachen dieser Zahl aus der Liste.
- (iii) Betrachte die kleinste, noch nicht verwendete Zahl aus der verbleibenden Liste und wiederhole Schritt (ii), solange $p^2 < N$ erfüllt ist.

MATHEMATICA eignet sich hervorragend zum sog. *Rapid prototyping*. Darunter versteht man schnelles Implementieren von neuen Ideen und Algorithmen, um sie auf Ihre Tauglichkeit zu beurteilen. Dies soll nun anhand des vorliegenden Algorithmus veranschaulicht werden.

- (a) Schreiben Sie eine MATHEMATICA-Routine `MyPrimes` (unter Verwendung von `Module`), die diesen Algorithmus implementiert, nur die obere Grenze N als Parameter hat und die Liste der Primzahlen kleiner oder gleich N zurück gibt.
- (b) Schreiben Sie eine Routine `MathPrimes`, die auf dem MATHEMATICA-Befehl `NextPrime` aufbaut, nur die obere Grenze N als Parameter hat und die Liste der Primzahlen kleiner oder gleich N zurück gibt.
- (c) Vergleichen Sie das Laufzeitverhalten von `MyPrimes` und `MathPrimes` unter Zuhilfenahme des Moduls:

```
In[1]:= CheckPrimes[f_, bound_] :=  
  Module[{r, t},  
    {t, r} = AbsoluteTiming@f[bound];  
    Print["running time: ", t];  
    Print["# of found primes: ", Length[r]];  
    Print["non-prime entries: ", Select[r, ! PrimeQ[#] &]]  
  ]
```

(Freiwillige) Zusatzaufgaben:

- (a) Wie können Sie Ihre Routine `MyPrimes` verbessern?
- (b) Machen Sie sich die Funktionsweise von `CheckPrimes` klar.
- (c) Schreiben Sie eine Routine, die auf dem MATHEMATICA-Befehl `Prime` aufbaut.

Hinweise:

- Beim Lösen der Aufgabe ist es durchaus erlaubt (jedoch nicht erforderlich), die Programmierkenntnisse aus der C++-Vorlesung unmittelbar in MATHEMATICA umsetzen.
- Hilfreiche MATHEMATICA-Befehle: `Append`, `Do`, `First`, `If`, `For`, `Join`, `Mod`, `Module`, `Range`, `Select`, `While`.